

1 **LOW POWER DIGITAL AUDIO DECODING/PLAYING SYSTEM**

2 **FOR COMPUTING DEVICES**

3 This application claims the benefit of provisional application serial No. 60/250,899, filed
4 on December 1, 2000, entitled "Low Power Digital Audio Decoding System for Computing
5 Devices" and provisional application serial No. 60/265,466, filed on January 30, 2001, entitled
6 "Low Power Digital Audio Decoding/Play System for Computing Devices".

7 **BACKGROUND OF THE INVENTION**

8 **1. Field of the Invention**

9 The present invention relates generally to portable devices (e.g., notebook computers) for
10 reproducing audio recordings, and more particularly, to low-power hardware and software for
11 decoding and reproducing compressed audio recordings in a variety of compression formats
12 from a variety of sources. While particular utility for the present application is in the
13 reproduction of MP3 digital audio files, especially for use with portable computers, other utilities
14 are contemplated herein.

15
16 **2. Description of Related Art**

17 Presently there exist various portable devices for replaying digital audio recordings that
18 have been compressed in accordance with one or more compressed audio digital recording
19 formats, e.g., MPEG (Moving Picture Experts Group) Audio Layer-3 (MP3), Windows® Media
20 Audio (WMA), and Advanced Audio Coding (AAC). To date, the most popular format has been
21 MP3, a compression scheme that results in about a 10:1 compression of the size of digital music
22 files. These devices can be divided into two classes, those which store the compressed digital
23 audio recordings in an electronic solid-state memory, and those which record the compressed

1 The prior art also includes silicon solutions that are dedicated function integrated circuits
2 (ICs) or incorporated into application-specific integrated circuits, or ASICs. These are usually
3 expensive solutions as the digital signal processor (DSP) required in a dedicated chip results in a
4 large, costly integrated circuit. One of the results is the use of a larger amount of PCB (printed
5 circuit board) space.

6 Further, the 15 to 20 MIPS (million instructions per second) decode engine known in the
7 art must be continuously running to generate the audio stream for the Codec. Additionally, the
8 dedicated decode engine needs to have the high-power-consuming hard disk drive (HDD)
9 continuously operating. These approaches are limited to functioning only with MP3
10 compression, thereby eliminating the opportunity to adapt the system to newly emerging music
11 compression algorithms, such as Microsoft's WMA or the music industry's proposed Secure
12 Digital Music Initiative (SDMI) for secure audio.

13 Dedicated silicon solutions known in the art employ a DSP that must constantly be
14 decoding the compressed audio files from a hard disk drive, which must therefore be constantly
15 reading the audio files. Such known methods require much power, resulting in a fast battery
16 discharge, (e.g., much faster than the possible 4 to 10 hours of desired use on a transoceanic
17 flight).

18 Thus, known hardware MP3 decoder and players requiring an IC implementation and a
19 hard disk drive being accessed non-stop are high in power consumption, difficult to upgrade, and
20 expensive.

21 The present invention provides a solution that is low in power consumption, can be
22 upgraded in the field for various music compression formats, is expected to cost no more than
23 half the cost of the currently available hardware implementation, and may be made capable of

playing up to hundreds of musical selections, while only having to access the HDD or CD-ROM less than 0.5% of the time.

SUMMARY OF THE INVENTION

It is becoming more and more desirable for mobile platform companies to add MP3 and other compressed audio player capability to their products, with low cost, while providing very long music playing time, and perhaps even a player that can be later upgraded to other compression formats by the owner. These mobile platform companies may also want to market differentiate their products within a very short development time frame.

The music playing solution of the present invention utilizes a special purpose circuit in combination with the mini-OS (operating system) software of the present invention. The present invention uses the embedded computing power of the standard CPU to perform the file decompression. Since today's CPUs with clock rates of 500 MHz to 1 GHz have at least an order of magnitude higher processing power than the real time DSP engines used in currently available MP3 player/decoders, these powerful CPU processors can often finish the decoding process in less than 10% of the available time. The CPU may then be set to idle by the present invention for more than 90% of the time, saving large amounts of power and thus greatly slowing the discharge of the battery and extending the useful time of the equipment under battery power on a single charge.

The present invention is unlike the real-time DSP engines known in the art, which require a constant data stream from the HDD, and which result in high power consumption, since the HDD is being accessed all the time. Using the technology of the present invention, the HDD may be accessed less than 0.5% of the time with a typical complement of memory, i.e., 128MB RAM. This results in a dramatic reduction in the rate at which power is dissipated from the

1 equipment battery. Further, minimal PCB changes are required for the present invention, thus
2 resulting in the quick adoption of new product features in PCs.

3 There are many possible music compression algorithms. Compression algorithms other
4 than MP3 include WMA, AAC, and the proposed SDMI. The software decompression
5 methodology of the present invention can be easily modified to decode any compression scheme,
6 or with a software installation process, all the various compression schemes. This flexibility
7 allows the adaptation to new and different algorithms, as they become popular, by permitting an
8 after-market upgrade of computers equipped with the present invention. Also, since this portion
9 of the present invention is a software system, new updates and/or algorithms may be downloaded
10 (e.g., from the Internet) to upgrade machines in the field, eliminating the necessity for consumers
11 to buy multiple players/decoders in order to listen to audio files having different compression
12 formats.

13 Thus, the present invention provides a low-cost, low power-consumption, long-battery-
14 life audio playing and decoding system, which may be used to play audio files of various
15 formats.

16 In one aspect, a computer system adapted to play audio files comprises a system CPU,
17 memory; at least one drive comprising compressed audio data residing in one or more audio
18 files, a play list software program for selecting and storing a play list comprising one or more of
19 the audio files, a first operating system adapted to control at least the system CPU and memory,
20 and a second operating system stored in BIOS and adapted to retrieve the play list and cause the
21 drive to read at least one audio file of the play list, to cause the system CPU to decompress the
22 compressed audio data of the file and provide decompressed audio data, and to cause the
23 decompressed audio data to be stored in memory.

1 In another aspect, a computer system adapted to play audio files comprises a drive
2 comprising at least one audio file, an audio controller, and an operating system stored in BIOS,
3 the operating system controlling the audio controller, so as to cause the audio controller to play
4 at least one audio file.

5 In a further aspect, a computer system adapted to play audio files comprises: compressed
6 audio data, a system CPU, an audio controller, a first operating system adapted to control at least
7 the system CPU, a second operating system controlling the audio controller and system CPU, so
8 as to cause the system CPU to decompress the compressed audio data, and a switch, the
9 activation of the switch causing the second operating system to boot.

10 In yet another aspect, a computer system adapted to play audio files comprises a system
11 CPU, memory, at least one drive comprising compressed audio data residing in one or more
12 audio files, a play list software program for selecting a play list comprising one or more of the
13 audio files, and an audio controller coupled to the system CPU, memory and drive. The audio
14 controller is adapted to cause the drive to read at least one audio file of the play list, to cause the
15 system CPU to decompress the compressed audio data of the file and thereby provide
16 decompressed audio data, and to cause the decompressed audio data to be stored in memory.

17 In process form, a method of playing audio files on a computer system comprises:
18 booting a first operating system; creating and storing a play list comprising a list of compressed
19 audio files residing on one or more drives of a computer system having at least a drive, a CPU,
20 and a memory; terminating the first operating system; booting a second operating system upon
21 activation by a switch; reading the play list; reading the compressed audio files from the drive
22 based on the play list; providing the compressed audio data to the CPU for decompressing the
23 data of the compressed audio file into decompressed audio data; storing the decompressed audio

1 data in memory; and retrieving the decompressed audio data from the memory for playing.

2 In another process form, a method of playing audio files on a computer system
3 comprises: reading compressed audio data from the drive of a computer system having at least a
4 drive, a CPU, and a memory; providing the compressed audio data to the CPU for
5 decompressing the compressed audio data, thereby providing decompressed audio data; and
6 storing the decompressed audio data in memory.

7 **BRIEF DESCRIPTION OF THE DRAWINGS**

8 Figure 1 is a block diagram representation an exemplary operational flow of one
9 embodiment of the present invention;

10 Figure 2 is a flow diagram of an exemplary power up of the mini-OS and initiation of the
11 player function, in one embodiment of the present invention;

12 Figure 3 is a block diagram of an exemplary audio player system consistent with one
13 embodiment of the present invention; and

14 Figure 4 is a block diagram of the internal portion of an exemplary special purpose
15 circuit, in relation to the other components that interface with it, in one embodiment of the
16 present invention.

17 **DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS**

18 The present invention comprises mini-OS (operating system) software and a hardware interface
19 between the South Bridge and Codec to play the musical selections (or other stored audio)
20 desired by the user. The mini-OS software of the present invention performs only those
21 functions and enables those elements of the portable computer that are needed, when they are
22 needed, to play the selected music, without performing all of the background functions
23 performed by the full system operating system, e.g., Windows®, and without accessing the

1 monitor circuitry and monitor screen of the portable computer. Additionally, the mini-OS of the
2 present invention only accesses the HDD when compressed files are being transferred to RAM.
3 Thus, it will be seen that the mini-OS software portion of the present invention performs both
4 power saving and file management functions when playing audio.

5 Figure 1 is a block diagram representation of the operational flow of the exemplary
6 software compressed audio player in one embodiment of the present invention.

7 The operational concept illustrated in Figure 1 is as follows:

8 1st: A browser, running on a full system operating system, e.g.,
9 Windows®, of the portable computer is initially used to download compressed
10 music files (for example 1000 songs) onto the PC hard disk drive (HDD) (2) (e.g.,
11 using 4 gigabytes of HDD space) at some time prior to the time at which the user
12 desires to use the portable computer as an audio player and a playlist is created,
13 comprising the songs the user desires to hear at a later time;

14 2nd: When the user desires to use the portable computer as an audio
15 player, once the desired music files are on the HDD, the user operates an audio
16 player on-switch to turn the portable computer fully on, boot up the entire
17 computer, load in the mini-OS of the present invention instead of the usual
18 Microsoft Windows® OS (the full system operating system is not opened) with the
19 power saving initialization subroutines and initializes only those portions of the
20 portable computer as necessary, and the file management subroutines initialize
21 the song play list or book generated in step 1, of a substantial number of songs,
22 for desired music listening under direction of the user;

23 3rd: The mini-OS software is then copied from the HDD (2) to RAM

1 (4), and then the first set of compressed files from the song play list is copied
2 from the HDD (2) to the system RAM (4) also using the mini-OS software of the
3 present invention. For example, in today's PC's 128 Mbytes is a typical system
4 RAM size, with the mini-OS software of the present invention taking about 8
5 Mbytes of the RAM, leaving approximately 120 Mbytes for use as a compressed
6 music memory (i.e., a cache or buffer, using system memory, dedicated memory,
7 or other memory). That 120 Mbytes represents about 2 hours of continuous
8 compressed music with a compression ration of 10:1, typical of MP3 files.
9 Similarly, in the case when flash media is used for MP3 storage, all or most of the
10 contents of the flash media card can be copied to the system RAM (4), thus
11 minimizing the access of the flash media reader and allowing for a more
12 responsive control over the MP3 files;

13 4th: The file management software of the present invention
14 sequentially delivers portions of the first music file to the CPU (6) where the
15 decode algorithm decompresses each file using the file management software of
16 the present invention stored in RAM (4). Once decoded, the PCM audio data is
17 transferred in one of three ways: the CPU delivers the PCM audio data to the
18 South Bridge (see Figure 3 (32)) FIFO buffer; the DMA in the South Bridge
19 transfers the data internally within the South Bridge to the FIFO buffer; or the
20 special purpose circuit transfers the data to the FIFO buffer from the LPC
21 interface. The FIFO buffer then sequentially feeds each piece of decoded music
22 to Codec (8) (also see Figure 3 (42)), through the special purpose circuit of the
23 present invention, where the decoded signal is converted from digital to analog.

1 Then the output signal from the Codec (8) is amplified (10) (also see Figure 3
2 (44)) to drive the speakers and/or headset (see Figure 3 (46)).

3 5th: While the final song of the first set from the play list is playing
4 from memory, the file management software of the present invention stored in the
5 RAM (4, 30) returns control to the 4th step to retrieve the next set of compressed
6 music files from the memory of the RAM, as determined by the earlier scripted
7 song play list developed in the 1st step. Thus, the 4th and 5th steps are repeated for
8 each set of compressed music files until the last music selection in the set plays.
9 At that point in time control returns to the 3rd step to load another set from the
10 play list, which is similarly played through the 4th and 5th steps. When the last
11 song is played from the overall play list of the 2nd step, or when the user turns off
12 the music player function, the operation of the player ceases.

13 The mini-OS power saving software of the present invention ensures that the CPU,
14 Peripheral Chips, HDD and other controllable system elements will be in idle state for the
15 highest percentage time possible. An interesting attribute of the solution offered by the present
16 invention is that the higher the MIPS (Million Instructions Per Second) capacity of the CPU, the
17 smaller percentage of time the CPU will spend performing the decode function. This means that
18 higher performance CPU's will demonstrate even lower power usage when playing compressed
19 music performances, thus saving even more battery power and further extending the length of
20 time that the battery maintains sufficient charge to power the portable computer.

21 The mini-OS monitors the audio control buttons (e.g., play, fast forward, rewind, pause,
22 scan, previous track, next track, first track, last track, fast forward/rewind while listening, audio
23 source/media select (e.g., HDD or CD), etc.) (see Figure 3 (48)) for user actuation through the

1 special purpose circuit (see Figure 3 (40)) of the present invention, and communicates user
2 requests to the mini-OS file management software of the present invention. Optionally, a small
3 LCD display (see Figure 3 (34)) can be connected to the special purpose circuit to provide visual
4 status indicators (e.g., Song #, Song titles, track #, Playtime & icons) under control of the mini-
5 OS display management subroutines.

6 The mini-OS power saving software of the present invention primarily manages the
7 usage of the CPU, and the MP3 storage devices such as CD, HDD, and flash media such as SD
8 (Secure Digital) cards, MMC (Multimedia Card), memory stick, and SMC (Smart Media Card),
9 while maintaining the rest of the system, including the memory, corelogic chipsets, in a fully on
10 and functional state. Secondary power saving is applied to other PC subsystems to minimize
11 power usage still further by putting them in an idle state.

12 For example, with a 500 MHz Pentium III CPU having about 225 MIPS of processing
13 power and the decode algorithm requiring about 15 MIPS, the CPU will be operating less than
14 10% of the time. The other 90-95% of the time the CPU will be in a standby mode that requires
15 only milliamps of current. Alternatively, the CPU can be run at a slower clock speed, which is
16 usually an option provided by most of today CPUs, such as the AMD's Athlon CPU. Similarly
17 the HDD is accessed during the time it takes to fill or refill the RAM. Thus, since the average
18 song takes about 4 minutes to play and the RAM holds about 30 songs for 120 Mbytes, and since
19 the HDD needs 1-5 seconds to spin up and only several seconds to load the song play list into
20 RAM, the total access time for the HDD may be 30 seconds out of 120 minutes of play time; a
21 ratio of 1:240, less than 0.5% of full power operating time. These factors add to the power
22 savings gained by using the mini-OS of the present invention instead of the full operating system
23 of the portable computer. The result of the overall power consumption of the present invention is

1 very low when the portable computer is in the music play mode, and that directly translates into
2 the battery maintaining a useful charge level for a much longer time than allowed by the prior
3 art. As those skilled in the art will recognize, the compressed music data of this invention may
4 reside on a hard disk, on other magnetic (e.g., tape) media, optical (e.g., CD-ROM) media, flash
5 media (e.g., SD cards, MMC, memory stick, SMC), or any other storage medium.

6 Figure 3 is a generalized overall block diagram of an exemplary system 31 consistent
7 with one embodiment of the present invention. The majority of the blocks in system 31 are
8 components known in the art and are generally included in all PC computers for producing
9 sound through the speaker of the computer. Shown here is a system clock 56, which, for
10 simplicity of Figure 3, is not shown connected to the various components that need a clock
11 signal. Additionally, CPU 26 is shown interfacing with North Bridge 28. In turn, North Bridge
12 28 interfaces with system RAM 30 and South Bridge 32. Then South Bridge 32 interfaces with
13 HDD 36 and CD-ROM 38. Typically South Bridge 32 also interfaces directly with Codec 42
14 through AC_link; however, in the exemplary system 31 shown, special purpose circuit 40 (see
15 discussion of Figure 4 below) is inserted between South Bridge 32 and Codec 42 to enable the
16 playing of compressed digital audio in conjunction with the mini-OS 80 of the present invention
17 from system RAM 30, without affecting the ability to play non-compressed analog audio. In this
18 configuration, the mini-OS 80 is stored in the BIOS, although those skilled in the art will
19 recognize that the mini-OS could alternatively be stored in its own ROM (either within special
20 purpose circuit 40 or external to it), a hard disk, or other media. Thus, AC_link₁ from South
21 Bridge 32 is coupled to special purpose circuit 40, which performs the decompression function
22 as necessary, and then provides any audio signals to Codec 42 via AC_link₂. Codec 42 then
23 performs the usual function on all signals received from special purpose circuit 40 and applies

switches 60 closed connecting AC-link₁ with AC_link₂ when the PC functions normally with the full system OS, and with switches 60 open when a system consistent with the present invention is employed. The LPC path is coupled to LPC interface. Switches 60 and AC_link₂ are coupled to state machine 64, while another port of state machine 64 is coupled, via bus 74, to the output of LPC interface 62, as well as register block 66, function key interface 68 and LCD interface 72. A second port of register block 66 is also coupled to a third port of state machine 64. Function keys 48 are coupled to function key interface 68, and LCD 34 is coupled to LCD interface 72. Also, function key interface 68 provides a signal to register block 66 when one of the function keys 48 is selected by the user. Audio player power switch 54, which is operated by the user in the second step discussed above, may be used to activate the PC to operate as described hereinabove. Switch 54 is shown connected to the DC voltage source of the portable computer and not to any particular block in Figure 4, since that connection varies depending on several factors controlled by the manufacturer of the computer on which an embodiment of the present invention is installed.

More specifically, the blocks within special purpose circuit 40 operate as follows:

LPC Interface

Special purpose circuit 40 includes LPC (Low Pin Count) interface 62 to interface with LPC controller 52 in South Bridge 32.

The LPC interface 62 is used to by CPU 26 to:

- (1) read the function key input registers in register block 66;
- (2) set the control register in register block 66 to control the AC97 Codec 42;
- (3) get the audio PCM (Pulse Code Modulation) data from the system memory (RAM 30); and

1 (4) perform clock throttling control.

2 The setting in the mode register of register block 66 controls the state of switches 60 to
3 switch the special purpose circuit 40 between the normal computer operation mode with
4 switches 60 closed (e.g., running Microsoft Windows® OS) and the mode of a system consistent
5 with the present invention, with switches 60 open (running the mini-OS) to play compressed
6 audio files.

7 **South Bridge AC97 Controller 50 interface (AC_Link₁ from host)**

8 During the normal computer operation mode, switches 60 are closed with the South
9 Bridge AC97 Controller 50 interface connected directly through, closed switches 60, to AC97
10 Codec 42 to generate audio output as if special purpose circuit 40 were not present. To play
11 compressed audio files, switches 60 are open when the mini-OS is running, and state machine 64
12 controls AC97 Codec 42.

13 **AC97 Codec interface (AC_Link₂ to AC97 Codec 42)**

14 When the computer is running under control of the mini-OS, switches 60 are open. State
15 machine 64 then controls the AC_link₂ in response to the settings of the register block 66 set by
16 the host (CPU 26) to generate the controls for AC97 Codec 42 (e.g., switching the sampling
17 frequency, controlling volume, sending the PCM data to the Codec 42, setting the Codec 42 to
18 the power saving mode or waking Codec 42 from the power saving mode).

19 **Function Key Input Interface 68**

20 Function key interface 68 receives the user selections from function keys 48 and stores
21 the selections in internal registers to be read by CPU 26.

22 **LCD interface 72**

23 LCD interface 72 is only necessary if LCD 34 is used to provide status information to the

1 user. The purpose, when used, is to show player status on low cost LCD 34 when the system
2 consistent with the present invention is used. Status of the audio track number of the selection
3 playing, status icons (e.g., Play) and other generic status icons may be programmed into the
4 system and displayed for any other purpose.

5 Operation Modes

6 (A) Normal Operation Mode:

7 When the PC is fully powered and running under the full system OS, the various
8 functions of special purpose circuit 40 are bypassed and switches 60 are closed, as discussed
9 above. In the normal mode, the computer system uses the South Bridge AC97 Controller 50 to
10 directly control the AC97 Codec 42 through the AC_link (in the Normal mode AC_link₁ and
11 AC_link₂ are the same since switches 60 are closed. The special purpose circuit does not
12 intercept or modify the AC_link signals.

13 (B) Compressed Audio Performance Mode:

14 When switch 54 has been closed, the system runs under the control of mini-OS, and
15 special purpose circuit 40 is empowered and runs in the compressed audio performance mode.
16 The South Bridge AC97 Controller 50 is isolated from the AC97 Codec 42 in this mode since
17 switches 60 are open.

18 In the compressed audio performance mode, the host (CPU 26) sets the internal registers
19 of register block 66 to control the data flow to the AC97 Codec 42, and to perform the various
20 power management functions.

21 A Power Saving Control Method in Compressed Audio Performance Mode

22 A flexible control method of the special purpose circuit 40 is provided to minimize the
23 system control cycles and power consumption in the performance mode. The system memory

(RAM 30) is used to pass most of the control commands to the special purpose circuit 40, instead of CPU 26, which minimizes the time that CPU 26 needs to access high speed external bus other than a standby level. This considerably reduces the power load on the portable computer battery in this mode.

CPU 26 also sets the system control memory registers in register block 66. State machine 64 bases operation on those register settings to obtain control words and PCM data automatically through the LPC interface 62. The control words in the system memory (RAM 30) are fetched into the internal registers, and the state machine 64 decodes the control words to determine if PCM or audio data is ready. If the audio data is ready, the state machine 64 continues to fetch the audio data and send it to the AC97 Codec 42. The control words in the system memory (RAM 30) can also be used to indicate the sampling frequency of the PCM data. So, the state machine 64 can set AC97 Codec 42 to the appropriate frequency before the PCM data is sent.

Those skilled in the art will recognize that a headphone or headset system may comprise further functionality than described hereinabove, e.g., a volume control, or the audio control buttons may be integrated thereto.

It should also be recognized that a special purpose circuit consistent with the invention may be integrated into a full-time compressed (and/or non-compressed) audio playing system capable of playing music regardless of the operation of the rest of the system. In this configuration, the special purpose circuit and mini-OS are provided, as well as a software driver for handling interrupts from the function buttons under Windows®. In this configuration, when the rest of the system is either fully on (S0) or in “sleep” (suspend to RAM or S3) mode, the system may be configured to begin execution of a custom or standard audio player, e.g., Music

1 before entering any sleep state. For example, a system cannot transition from state S2 to S4, nor
2 from state S4 to S2. It must first return to S0, from which it can enter the next sleep state.
3 Because a system in an intermediate sleep state has already lost some operating context, it must
4 return to the working state to restore that context before it can make an additional state
5 transition.

6 Referring now to Figure 2, in conjunction with Figure 3, an exemplary sequence 200 for
7 the power up of the mini-OS and initiation of the player function, in one embodiment of the
8 present invention, is illustrated. As stated above, at some time prior to the initiation of the audio
9 player function of a PC equipped with the present invention, the user downloads (not shown in
10 Figure 2) the audio files of interest to the HDD 36 or burns a CD-ROM that is placed in the CD-
11 ROM drive 38 for use with the audio player feature of the present invention. As shown, at step
12 201, the sequence 200 begins when the user presses either an audio player power switch 54 or
13 the computer's main power switch (not shown in Figure 3), to turn the system on. A
14 determination is then made, at step 202, whether the computer is to boot in normal operation
15 mode or compressed audio performance mode. This determination is typically made in the
16 BIOS, based on whether the computer's power switch or an audio player power switch 54 was
17 used to turn on the computer, although those skilled in the art will recognize that this
18 determination could alternatively be made by an application program or an operating system that
19 provides such capability (e.g. Windows® 98). If the computer's power switch was used to turn
20 on the computer, then the system boots to normal operation mode, at step 203, and the normal
21 operating system (e.g., Windows® 98) is loaded into system RAM 30 and executed. If an audio
22 player power switch 54 was used to turn on the computer, the mini-OS is loaded into system
23 RAM 30, at step 204. At step 203, the mini-OS initializes the system components including one

1 or more of the North Bridge 28, South Bridge 32, special purpose circuit 40, hard drive 36, CD-
2 ROM drive 38, codec 42, and CPU 26.

3 Since no audio decompression request will be pending upon system initialization (i.e., the
4 memory buffer is not full), which determination is made at step 208, the system waits for input
5 from one of the function keys 48, at step 207, until one of the function keys 48 is pressed, at
6 which point the appropriate function is executed and the LCD display updated, as appropriate, at
7 step 206. If the command includes a request from the user to play audio, an audio

8 decompression request will be pending at this time, which determination is made at step 208.

9 Since no compressed audio file(s) are in system memory 30 upon the initial request to play
10 audio, which determination is made at step 209, the compressed audio file(s) are read from the
11 HDD 36 and/or CD-ROM drive 38 and loaded into system memory 30, at step 210. After the
12 compressed audio files are loaded into system memory at step 210, or if the audio file(s) are
13 already in system memory, which determination is made at step 209, the audio files are then
14 decompressed, at step 211, using the system CPU 26. DMA transfer(s) to the codec 42 are
15 initialized for the decompressed audio data, at step 212, and then the output signal from the
16 Codec 42 is amplified (not shown in Figure 2) by the amplifier 44 to drive the speakers and/or
17 headset 46. After the DMA transfer(s) are initialized, at step 212, control loops back to step 208,
18 to determine whether an audio decompression request is pending.

19 It should be recognized by those skilled in the art that, although the above-described
20 embodiments utilize a hardware-based OS selection (i.e., pressing main power button boots to
21 Windows®, while pressing audio control button boots to mini-OS), other OS selection methods
22 are contemplated, as well. Such selection methods include, e.g., using a batch file or other
23 scripting or software-based method to shut down a first OS and boot to the second OS. Those

1 skilled in the art will also recognize that the mini-OS of the present invention could conceivably
2 be implemented as part of a larger OS (e.g., a GUI-based OS, such as Windows®, LINUX, etc.)
3 or as a software component named something other than an “operating system”, (e.g., a “driver”,
4 an “algorithm”, a “script”, “code”, a “program”, an “executable”, a “routine”, a “subroutine”, a
5 “utility”, etc.), instead of being implemented as an entirely separate operating system. Such
6 embodiments are contemplated to be within the scope of the present invention.

7 Although the present invention has been described in terms of the exemplary
8 embodiments provided herein, it is to be understood that such disclosure is purely illustrative and
9 is not to be interpreted as limiting. Consequently, without departing from the spirit and scope of
10 the invention, various alterations, modifications, and/or alternative applications of the invention
11 will, no doubt, be suggested to those skilled in the art after having read the preceding disclosure.

12 Accordingly, it is intended that the following claims be interpreted as encompassing all
13 alterations, modifications, or alternative applications as fall within the true spirit and scope of the
14 invention.

15
16 What is claimed is:
17